**國立雲林科技大學電機工程系**

**碩士論文**

**Department of Electrical Engineering**

**National Yunlin University of Science & Technology**

**Master Thesis**

**機器學習 專案作業一**

**Machine Learning Project Assignment 1**

**M11212027林彥廷**

**M11212025吳冠聰**

**M11212085楊勝瑋**

**指導教授：許中川 教授**

**中華民國113年4月  
April 2024**

摘要

分類和回歸是機器學習中兩個重要的任務，它們在各種領域中都有廣泛的應用。本文探討了分類和回歸技術來進行預測的主題。我們以波士頓房價資料集（Boston Housing Price）為例，該資料集包含了波士頓地區不同地段的房屋價格以及相關特徵。在分類任務中，我們可以利用房屋的特徵來預測它們的類別，例如人均犯罪率、每套住宅的平均房間數、一氧化氮濃度。而在回歸任務中，我們則可以利用這些特徵來預測房屋的實際價格。本文介紹了分類和回歸任務的基本概念，並探討了在波士頓房價數據集上應用分類和回歸技術的方法。通過對波士頓房價資料集的分析和實驗結果，我們可以更好地理解和應用分類和回歸技術來解決現實世界中的問題。

關鍵字：深度學習、神經網路、分類及迴歸預測

# 緒論

## 研究動機

第一題:MNIST資料集是一個包含手寫數字圖像的經典數據集，被廣泛用於機器學習和深度學習研究中，此動機在於進一步探索和挖掘MNIST資料集的價值，深入了解手寫數字圖像的特徵和分佈情況，此外，基於對MNIST資料集的研究，可以擴展到更廣泛的圖像識別和分類領域，為人們提供更智能、更高效的圖像識別技術。

第二題: Boston Housing Price資料集是一個經典的房地產數據集，包含了多個影響房價的特徵，如人均犯罪率、地理位置、環境等。此動機在於深入探索這個數據集，分析不同特徵對房價的影響程度及作用機制，進一步理解房地產市場的運作規律。通過建立準確的房價預測模型，可以幫助投資者和開發商做出更明智的投資決策，並為家庭購房提供參考。

第三題:利用各種數據預測成人收入水平，進一步探索和理解影響個人收入的各種因素及其複雜關係。通過分析種族、教育程度、職業類型、性別等多個方面的數據，尋找與收入密切相關的關鍵因素，並建立準確的預測模型。這不僅有助於個人更好地了解自身收入水平的形成機制，還能為政府制定更有效的收入分配政策、企業進行人力資源管理、社會組織開展精準扶貧等提供科學依據。

## 研究目的

第一題:探索和分析MNIST資料集在影像分類領域的應用和效能，透過深入研究MNIST資料集，我們的目標是評估不同的影像分類演算法和模型在處理手寫數位影像時的效果，並比較它們的性能和準確性。透過研究MNIST資料集，我們也希望深入理解影像分類問題的挑戰和解決方案，探索改進現有演算法和開發新模型的可能性。

第二題:探索Boston Housing Price資料集的形成機制與影響因素，並建立準確的房價預測模型，以提供房地產投資者、開發商和居民購房者的決策支持。通過對波士頓地區房價的多角度分析，包括房屋特徵、地理位置、社區設施、經濟發展等因素，以深入了解影響房價的關鍵因素及其作用機制。同時，將收集並整理大量的波士頓房地產市場相關數據，運用統計分析方法和機器學習技術建立預測模型，以實現對未來房價變動的準確預測，為市場參與者提供可靠的投資建議和購房指導，促進波士頓房地產市場的穩健發展。

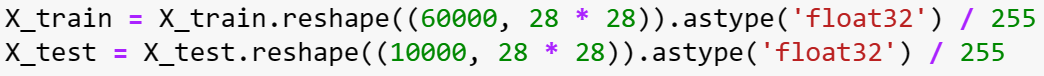
第三題:通過收集和分析各種可能影響收入的數據，包括性別、教育背景、職業類型、種族等因素，以探索其與收入之間的關係和影響程度。進一步，將運用機器學習、統計建模等技術，建立預測模型，從而實現對個人收入水平的準確預測。該研究旨在提高對收入分配不平等、貧富差距等社會問題的理解，並為政府制定相應政策、企業進行人力資源管理、社會組織進行公益活動等提供科學依據和建議。

# 方法

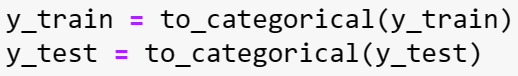
## 2.1 實驗一

第一個實驗資料集使用MNIST數據集，實驗model使用Sequential model，添加兩個全連接層。第一層有1000個神經元，啟動函數為ReLU。輸入形狀為28 \* 28，因為圖片的大小是28 \* 28像素。第二層有10個神經元，啟動函數為softmax，用於多類別分類。使用rmsprop優化器，categorical\_crossentropy作為損失函數，並設置metrics為accuracy來編譯模型。

### 2.1.1前置處理



將數據正規化，將像素值縮放到0到1的範圍內，以便更好地訓練模型。



對標籤進行one-hot編碼，將其轉換為二進制矩陣，以便模型能夠更好地理解標籤。

### 2.1.2實驗



接下來將驗證集與訓練集做訓練，batch\_size = 128，Epoch設定為5。

下面是該訓練的損失曲線、準確率曲線:

|  |  |
| --- | --- |
|  |  |

根據左圖我們可以觀察到:

由於訓練和驗證損失曲線幾乎重合,且都能夠收斂到較低的損失值，所以這組損失曲線表明模型學習過程很順利,能夠很好地擬合訓練數據。

根據右圖我們可以觀察到:

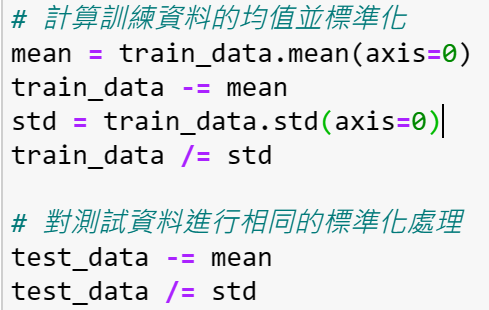
由於訓練和驗證準確率如此接近且達到很高的水平,沒有出現驗證準確率在後期下降的跡象，這組準確率曲線表明該模型能很好地擬合訓練數據。

## 2.2 實驗二

第二個實驗資料集是建立一個預測波士頓房價的神經網絡模型，並將它進行訓練。該模型包含兩個隱藏層，每個隱藏層有 64 個節點，並且使用了ReLU激活函數。

模型中使用了均方誤差（MSE）作為損失函數，以及使用RMSprop 作為優化器，並且還使用了平均絕對誤差（MAE）作為評估指標。

### 2.2.1前置處理



將資料進行標準化處理

### 2.2.2實驗



接下來將驗證集與訓練集做訓練，batch\_size = 16，Epoch設定為80。

下面是該訓練的損失曲線、準確率曲線:

|  |  |
| --- | --- |
|  |  |

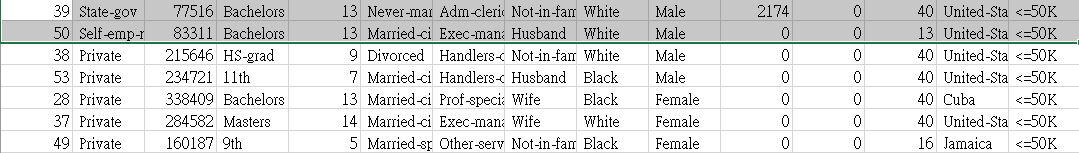
此模型是判斷MAE，所以曲線下降，代表模型訓練是成功的偏差會越來越小。

# 實驗

## 3.1資料集

資料集簡介，Barry Becker 從 1994 年人口普查資料庫中擷取資料。使用以下條件提取一組相當乾淨的記錄： ((AAGE>16) && (AGI>100) && (AFNLWGT>1)&& (HRSWK>0)) 預測任務是確定一個人的年收入是否超過 50K年。其中有15種特徵分別是age 年齡、workclass 工作類別、fnlwgt 人口普查樣本的最終權重、education 教育程度、education-num 受教育年數、marital-status 婚姻狀況、occupation 職業、relationship 與家庭關係、race 種族、sex 性別、capital-gain 資本收益、capital-loss 資本損失、hours-per-week 每週工作時數、

native-country 國籍、income 收入。workclass 工作類別、education 教育程度、marital-status 婚姻狀況、occupation 職業、relationship 與家庭關係、race 種族、sex 性別、native-country 國籍這些資料型態是字元所以需要經過資料前處理。

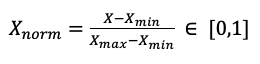


## 3.2前置處理

名目型欄位值轉換成數字，由於上述所提到的變數型態不一致的問題，我是使用sklearn.preprocessing裡面的LabelEncoder，一個 Python 中常用的機器學習庫中的一個模組，用於數據預處理（data preprocessing）可以幫助你在訓練機器學習模型之前，對數據進行必要的轉換、縮放、正規化等操作，以改善模型的性能或提高學習速度。可以看到以下原本為字元的資料已經被投影到參數上了。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |

欄位值正規化至0~1之間，假如沒有將資料做正規化的話訓練出來的loss會十分巨大，進而導致模型訓練出來效果不佳，驗證資料的正確率也不會提升。以下是正規化的數學式子。



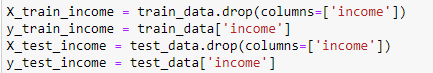
(原始資料-特徵最小)/(特徵最大-特徵最小)，還有測試集的資料也要做相同處理，切記要用訓練集得最大跟最小，因為測試前是不知道測試集資料特徵的最大跟最小值的。

接下來是資料遺失的部分，我使用的是isnull去查詢發現並未有資料遺失的狀況。

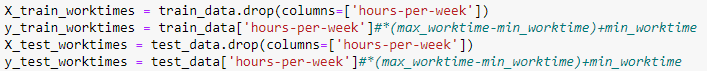
|  |  |
| --- | --- |
|  |  |

類別標記可以使用則是使用個別代表的英文使用基本的python語法，將個別欄位特徵用英文單字命名，可以更方便使用drop直接將欄位裡的資料移除跟直接提取出來，特徵跟訓練所需資料。方便後面需要訓練的目標更改像是:

分類預測：預測收入之類別 >50K 或 <=50K



迴歸預測：預測hours-per-week數值型欄位的數值。



## 3.3實驗設計

本實驗使用Python語言及Keras/Tensorflow實作前饋式神經網路預測類別與數值，使嗽得平台是Anaconda裡的Jupyter Notebook，從UCI ML Repository下載Adult Data Set，adult.data當作訓練資料集、adult.test當作測試資料集。

### 3.3.1分類預測：預測收入之類別 >50K 或 <=50K

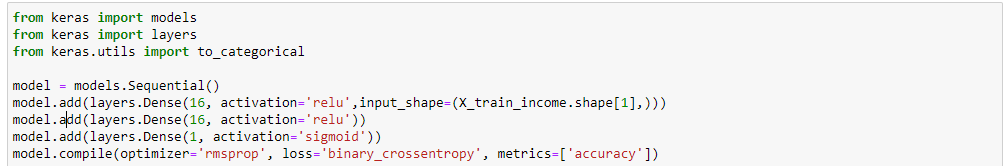
 本實驗model使用是Sequential model，層疊式神經網絡（feedforward neural networks）將其編譯成二元分類問題的模型。總共使用了3個全連接層(hidden layer)，

第一層有 16 個神經元，Activation Functions為 relu，輸入X\_train\_income.shape[1]；

第二層也有 16 個神經元，Activation Functions為 relu；

最後一層是一個具有 sigmoid Activation Functions的單一神經元。

用於二元分類問題的輸出。使用 rmsprop 優化器，binary\_crossentropy作為損失函數，並指定評估指標為準確率（accuracy）。



接下來將訓練集其他特徵(除了收入)跟只有訓練集收入放入此model做訓練，

batch\_size = 16，Epoch設定為80 驗證集則是放入vaildation\_data 也是一樣，將測試集其他特徵(除了收入)跟測試集收入，verbose=1則是將每次訓練的結果顯示出來。

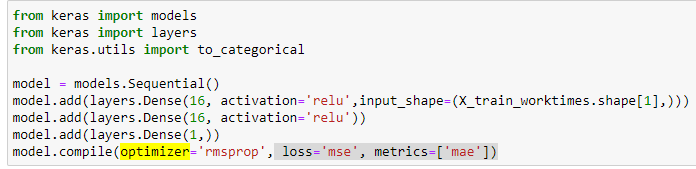
### 3.3.2迴歸預測：預測hours-per-week數值型欄位的數值。

本實驗model使用是Sequential model，層疊式神經網絡（feedforward neural networks）將其編譯成迴歸問題的模型。共使用了3個全連接層(hidden layer)分別有，第一層有 16 個神經元，Activation Functions為 relu，輸入X\_train\_worktimes.shape[1] ；

第二層也有 16 個神經元，Activation Functions為 relu；

最後一層則不使用Activation Functions用於迴歸問題的輸出。

使用了 rmsprop 優化器，均方誤差（MSE）作為損失函數，並且指定了平均絕對誤差（MAE）作為評估指標。



接下來訓練集其他特徵(除了每週工時)跟只有每週工時放入此model做訓練，

batch\_size = 16，Epoch設定為80 驗證集則是放入vaildation\_data 也是一樣，將測試集其他特徵(除了每週工時)跟每週工時，verbose=1則是將每次訓練的結果顯示出來。





## 3.4實驗結果

### 3.4.1分類預測結果：預測收入之類別 >50K 或 <=50K

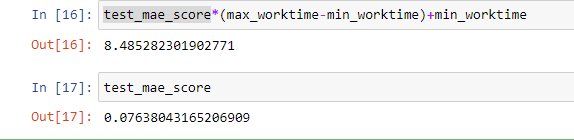
|  |  |
| --- | --- |
|  |  |

### 3.4.2迴歸預測結果：預測hours-per-week數值型欄位的數值。

因為此實驗有將hours-per-week經過正規化，因為此模型是判斷MAE，所以曲線下降，代表模型訓練是成功的偏差會越來越小。

|  |  |
| --- | --- |
|  |  |

，Max工作時間是99，Min工作時間則是1，所以實際上來說還有偏差8.48hr。

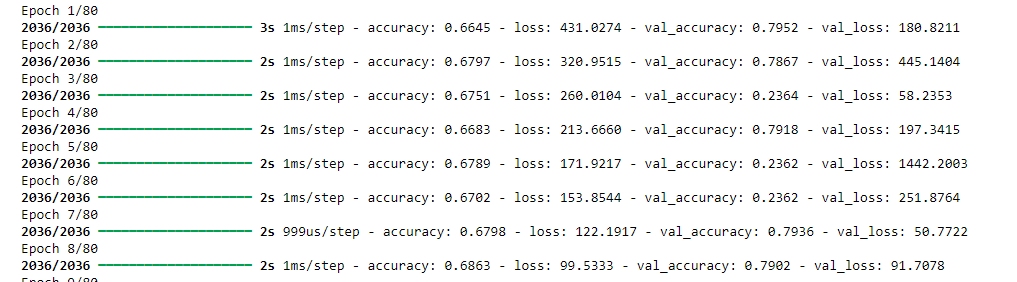


# 結論

## 4.1探討問題:

### 4.1.1資料正規化

假如資料沒有經過正規化直接下去做model\_training的話。訓練結果loss會很大，導致化出來的圖形準確率不會提升，假如不看前面loss大的值，收斂速度也會比起正規化後來得慢。還有測試集的準確率也會浮動的很誇張。



|  |  |
| --- | --- |
|  |  |

### 4.1.2 epoch過小

假如今天epoch 只設定5去訓練的話，可以看到還不能確定值是否收斂完成。還有正確率的浮動也是非常明顯，有此可以看出正確epoch對於實驗也是十分重要的

|  |  |
| --- | --- |
|  |  |

### 4.1.3 batch\_size過小

資料訓練時間會變得很長，運算量變得過於龐大。因為每做一次訓練就必須將值迴傳來更新類神經網路。

### 4.1.4 使用adam作為optimizer訓練結果比較

|  |  |
| --- | --- |
|  |  |

Adam val loss會慢慢與training loss逐漸分開，代表訓練模型已到了overfit的階段，可以停下實驗，rmsprop val loss會與training loss 保持一致，所以不能保證實驗室是否需要繼續進行。猜測其adam因為給予動量梯度下降更多，所以導致提早到達overfit的階段。

## 4.2 實驗心得:

此次實驗讓我更了解應該如何設計自己的AI訓練模型，然後各種不同的超參數應該如何調整。還有對於最後一層hidden layer的安排其實就是決定，是要做迴歸預測跟分類預測。

資料前處理對於實驗的結果影響十分得巨大，所以要先處理好資料去做模型訓練，是一件十分重要的事情。還有老師上課中提到的，將資料整理後和隱藏層做全連接，是AI 訓練一開始就要安排好的。最初開始時一直搞不懂為何程式會出問題。後面發現沒有將模型的input\_shape設定好所導致的。

# 參考文獻

## *資料正規化(Normalization)與標準化(Standardization)*. (n.d.). Https://Ithelp.Ithome.Com.Tw/Articles/10293893.